

1 This listing of claims will replace all prior versions, and listings, of claims in the
2 application:

3

4 **Listing of Claims:**

5 Claim 1 (Currently amended): A method for generating a dump file, the
6 method comprising:

7 a. generating a minidump file that does not include all operating system
8 data by gathering at least:

- 9 i. thread information for at least one running thread,
- 10 ii. context information for the thread,
- 11 iii. callstack information for the thread,
- 12 iv. process information for a process in which the thread is running,

13 and

14 v. information identifying a reason comprising one of the following
15 reasons: callstack fault, processor fault, and application program fault, for
16 generating the minidump file; and

17 b. storing the minidump file to a storage medium.

18 Claim 2 (Currently amended): The method as recited in Claim 1, further
19 comprising determining when to generate the minidump file.

20 Claim 3 (Currently amended): The method as recited in Claim 1, wherein
21 generating the minidump file further includes gathering processor information
22 about at least one processor.

23

24

25

1 Claim 4 (Currently amended): The method as recited in Claim 2, wherein
2 determining when to generate the minidump file further includes determining that
3 an exception has occurred.

4 Claim 5 (Currently amended): The method as recited in Claim 1, wherein
5 the minidump file does not include data stored in global initialized memory.

6 Claim 6 (Currently amended): The method as recited in Claim 1, wherein
7 the minidump file does not include data stored in uninitialized memory.

9 Claim 7 (Currently amended): The method as recited in Claim 1, wherein
10 the minidump file does not include executable instructions used by a processor to
11 execute a program.

12 Claim 8 (Currently amended): The method as recited in Claim 1, wherein
13 the minidump file is a kernel minidump file associated with an operating system
14 and the at least one running thread is the single thread which encountered an
15 exception.

17 Claim 9 (Previously presented): The method as recited in Claim 8, wherein
18 the callstack information includes kernel stack information.

19 Claim 10 (previously presented): The method as recited in Claim 1,
20 wherein the process information identifies a process that initiated the thread.

1 Claim 11 (Previously presented): The method as recited in Claim 1, further
comprising:

2 allocating a buffer space in memory during an initialization process,
3 wherein the buffer space is suitable for storing the gathered information; and

4 reserving space on the storage medium suitable for writing the contents of
5 the buffer space.

6
7 Claim 12 (Currently amended): The method as recited in Claim 11,
wherein generating the minidump file further includes initially storing the thread
information, the context information, the callstack information, the process
information, and the information identifying the reason for generating the
minidump file to the buffer space, and then copying the minidump file from the
buffer space to the storage medium ~~as a minidump file~~.

12
13 Claim 13 (Previously presented): The method as recited in Claim 12,
further comprising upon re-initialization, after having stored the minidump file to
the storage medium, accessing the minidump file on the storage medium and using
at least a portion of the minidump file to further understand an exception that was
at least one reason for generating the minidump file.

17
18 Claim 14 (Currently amended): The method as recited in Claim 1, wherein
the minidump file is a user minidump file associated with at least one non-
operating system program.

21
22 Claim 15 (Currently amended): The method as recited in Claim 1, wherein
generating the minidump file further includes gathering callstack information for
all running threads.

1 Claim 16 (Previously presented): The method as recited in Claim 15,
2 wherein the callstack information includes a user callstack.

3 Claim 17 (Currently amended): The method as recited in Claim 1, wherein
4 generating the minidump file further includes gathering processor context
5 information for all running threads.

6 Claim 18 (Currently amended): The method as recited in Claim 1, wherein
7 generating the minidump file further includes gathering a listing of loaded
8 modules for a faulting application program.

9 Claim 19 (Currently amended): The method as recited in Claim 1, wherein
10 the minidump file is a directory indexed file that uses relative virtual addresses
11 (RVAs).

13 Claim 20 (Currently amended): A computer-readable medium having
14 computer-executable instructions for causing at least one processor to perform acts
15 comprising:

16 gathering minidump file information that does not include all operating
17 system data but does include at least thread information for at least one running
18 thread, context information for the thread, callstack information for the thread,
19 process information for the process in which the thread is running, and
20 information identifying a reason comprising one of the following reasons:
21 callstack fault, processor fault, and application program fault, for generating the
22 minidump file; and ~~generating a dump file using the dump file information.~~

23 Claim 21 (Currently amended): The computer-readable medium as recited
24 in Claim 20, wherein generating the minidump file further includes storing the
25 dump file to a storage medium.

1 Claim 22 (Currently amended): The computer-readable medium as recited
2 in Claim 20, wherein gathering the minidump file information further includes
3 gathering processor information about at least one processor.

4

5 Claim 23 (Currently amended): The computer-readable medium as recited
6 in Claim 20, having further computer-executable instructions for causing the at
7 least one processor to perform acts comprising determining when to generate the
minidump file.

8

9 Claim 24 (Currently amended): The computer-readable medium as recited
10 in Claim 20, wherein the minidump file does not include data stored in global
11 initialized memory.

12

13 Claim 25 (Currently amended): The computer-readable medium as recited
14 in Claim 20, wherein the minidump file does not include data stored in
15 uninitialized memory.

16

17 Claim 26 (Currently amended): The computer-readable medium as recited
18 in Claim 24 wherein the minidump file does not include executable instructions used
19 by the at least one processor to execute a program.

20

21 Claim 27 (Currently amended): The computer-readable medium as recited
22 in Claim 20, wherein the minidump file is a kernel minidump file associated with
23 an operating system and the at least one running thread is the single thread which
24 encountered an exception.

1 Claim 28 (Previously presented): The computer-readable medium as
2 recited in Claim 20, wherein the callstack information includes kernel stack
3 information.

4 Claim 29 (Previously presented): The computer-readable medium as
5 recited in Claim 20, wherein the process information identifies a process that
6 initiated the thread.

7 Claim 30 (Currently amended): The computer-readable medium as recited
8 in Claim 20, further comprising computer-executable instructions for causing the
9 at least one processor to perform acts comprising:

10 allocating a buffer space in memory during an initialization process,
11 wherein the buffer space is suitable for storing the mini dump file information; and
12 reserving space on a storage medium drive suitable for writing the contents
13 of the buffer space.

14 Claim 31 (Currently amended): The computer-readable medium as recited
15 in Claim 30, wherein generating the minidump file further includes initially
16 storing the thread information, the context information, the callstack information,
17 the process information, and the information identifying the reason for generating
18 the mini dump file to the buffer space, and then copying the minidump file from
19 the buffer space to the storage medium ~~as a minidump file~~.

1 Claim 32 (Previously presented): The computer-readable medium as
2 recited in Claim 31, further comprising computer-executable instructions for
3 causing the at least one processor to perform acts comprising, upon re-
4 initialization after having stored the minidump file to the storage medium,
5 accessing the minidump file on the storage medium and using at least a portion of
6 the minidump file to further understand an exception that was at least one reason
7 for generating the minidump file.

8 Claim 33 (Currently amended): The computer-readable medium as recited
9 in Claim 20, wherein the minidump file is a user minidump file associated with at
10 least one non-operating system program.

11 Claim 34 (Currently amended): The computer-readable medium as recited
12 in Claim 20, wherein gathering the minidump file information further includes
13 gathering callstack information for all running threads.

14 Claim 35 (Previously presented): The computer-readable medium as
15 recited in Claim 34, wherein the callstack information includes a user callstack.

16 Claim 36 (Currently amended): The computer-readable medium as recited
17 in Claim 20, wherein gathering the minidump file information further includes
18 gathering processor context information for all running threads.

19 Claim 37 (Currently amended): The computer-readable medium as recited
20 in Claim 20, wherein gathering the minidump file information further includes
21 gathering a listing of all loaded modules for the faulting application program.

1 Claim 38 (Currently amended): The computer-readable medium as recited
2 in Claim 20, wherein the minidump file is a directory indexed file that uses
3 relative virtual addresses (RVAs).

4 Claim 39 (Currently amended): An apparatus comprising:
5 memory;
6 a data storage drive configured to write data files to at least one data storage
7 medium; and
8 at least one processor operatively coupled to the memory and the data
9 storage drive and configured to:

10 a. generate a mini dump file that does not include all operating
11 system data by gathering in the memory at least:
12 i. thread information for at least one running thread,
13 ii. context information for the thread,
14 iii. callstack information for the thread,
15 iv. process information for the process in which the thread is
16 running, and
17 v. information identifying a reason comprising one of the
18 following reasons: callstack fault, processor fault, and application
19 program fault, for generating the mini dump file, and
20 b. store the dump file to the storage medium.

21 Claim 40 (Currently amended): The apparatus as recited in Claim 39,
22 wherein the at least one processor is further configured to determine when to
23 generate the minidump file.
24
25

1 Claim 41 (Currently amended): The apparatus as recited in Claim 39,
2 wherein the at least one processor is further configured to gather processor
3 information about the at least one processor and include the processor information
4 in the minidump file.

5 Claim 42 (Currently amended): The apparatus as recited in Claim 40,
6 wherein the at least one processor is further configured to determine when to
7 generate the minidump file based on an exception.

8 Claim 43 (Currently amended): The apparatus as recited in Claim 39,
9 wherein the minidump file does not include data stored in global initialized
10 memory.

11 Claim 44 (Currently amended): The apparatus as recited in Claim 39,
12 wherein the minidump file does not include data stored in uninitialized memory.
13

14 Claim 45 (Currently amended): The apparatus as recited in Claim 39 wherein
15 the minidump file does not include executable instructions used by the at least one
16 processor to execute a program.
17

18 Claim 46 (Currently amended): The apparatus as recited in Claim 39,
19 wherein the minidump file is a kernel minidump file associated with an operating
20 system and the at least one running thread is the single thread which encountered
21 an exception.
22

23 Claim 47 (Previously presented): The apparatus as recited in Claim 39,
24 wherein the callstack information includes kernel stack information.
25

1 Claim 48 (Previously presented): The apparatus as recited in Claim 39,
2 wherein the process information identifies a process that initiated the thread.

3 Claim 49 (Previously presented): The apparatus as recited in Claim 39,
4 wherein the at least one processor is further configured to:

5 allocate a buffer space in the memory during an initialization process; and
6 reserve space on the storage medium drive suitable for writing the contents
7 of the buffer space.

8 Claim 50 (Currently amended): The apparatus as recited in Claim 49,
9 wherein the at least one processor is further configured to:

10 generate the minidump file by initially storing the thread information, the
11 context information, the callstack information, the process information, and the
12 information identifying the reason for generating the dump file to the buffer space,
13 and then copying the minidump file from the buffer space to the storage medium
14 as a minidump file.

15 Claim 51 (Previously presented): The apparatus as recited in Claim 50,
16 wherein the at least one processor is further configured to, upon re-initialization
17 after having stored the minidump file to the storage medium, access the minidump
18 file on the storage medium and use at least a portion of the minidump file to
19 further understand an exception that was at least one reason for generating the
20 minidump file.

21 Claim 52 (Currently amended): The apparatus as recited in Claim 39,
22 wherein the minidump file is a user minidump file associated with at least one
23 non-operating system program.

1 Claim 53 (Currently amended): The apparatus as recited in Claim 39,
2 wherein the at least one processor is further configured to gather callstack
3 information for all running threads as part of the minidump file.

4 Claim 54 (Previously presented): The apparatus as recited in Claim 53,
5 wherein the callstack information includes a user callstack.

6 Claim 55 (Currently amended): The apparatus as recited in Claim 39,
7 wherein the at least one processor is configured to gather processor context
8 information for all running threads as part of the minidump file.

9 Claim 56 (Currently amended): The apparatus as recited in Claim 39,
10 wherein the at least one processor is configured to gather a listing of all loaded
11 modules for a faulting application program as part of the minidump file.

12 Claim 57 (Currently amended): The apparatus as recited in Claim 39,
13 wherein the minidump file is a directory indexed file that uses relative virtual
14 addresses (RVAs).

15
16 Claims 58-66 (Canceled)

17 Claim 67 (Currently amended): The method as recited in Claim 1, further
18 comprising providing the minidump file to at least one external device.

19 Claim 68 (Currently amended): The method as recited in Claim 12, upon
20 system re-initialization, transferring the minidump file from the storage medium to
21 at least one external device.

1 Claim 69 (Currently amended): The method as recited in Claim 1, wherein
2 generating the dump file further includes gathering a list of loaded modules.
3

4 Claim 70 (Currently amended): The computer-readable medium as recited
5 in Claim 20, having further computer-executable instructions for causing the at
6 least one processor to perform acts comprising providing the minidump file to at
7 least one external device.

8 Claim 71 (Currently amended): The computer-readable medium as recited
9 in Claim 30, having further computer-executable instructions for causing the at
10 least one processor to perform acts comprising, upon system re-initialization,
11 transferring the minidump file from the storage medium to at least one external
12 device.

13 Claim 72 (Currently amended): The computer-readable medium as recited
14 in Claim 20, wherein gathering the minidump file information further includes
15 gathering a list of loaded modules.

16 Claim 73 (Currently amended): The apparatus as recited in Claim 39,
17 wherein the at least one processor is further configured to provide the minidump
18 file to at least one external device.

19 Claim 74 (Currently amended): The apparatus as recited in Claim 49,
20 wherein the at least one processor is further configured to, upon system re-
21 initialization, transferring the minidump file from the storage medium to at least
22 one external device.

1 Claim 75 (Currently amended): The apparatus as recited in Claim 39,
2 wherein the at least one processor is further configured to gather a list of loaded
2 modules as part of the minidump file.

3

4 Claims 76-77 (Canceled)

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25